

Trường Đại học Công nghệ - ĐHQGHN

Khoa Công nghệ thông tin

BÀI TẬP LỚN: PHÂN TÍCH & THIẾT KẾ HƯỚNG ĐỐI TƯỢNG

Giảng viên: Đặng Đức Hạnh



SOFTWARE ARCHITECTURE DOCUMENT

HỆ THỐNG QUẢN LÝ KHÁCH SẠN EZCLOUD

Ngày: 26/03/2023

Chuẩn bị bởi: Nhóm sinh viên: Nguyễn Đức Thiện, Nguyễn Trọng Linh, Tô Lâm Sơn, Mai Tú Phương, Phan Xuân Bảo

Mục lục

Lịch sử sửa đổi	3
1. Giới thiệu.....	4
1.1. Mục đích.....	4
1.2. Phạm vi.....	5
1.3. Thuật ngữ, Từ đồng nghĩa và Từ viết tắt	5
1.4. Tài liệu tham khảo.....	5
1.5. Tổng quan.....	6
2. Biểu diễn kiến trúc	6
2.1. Khung nhìn Logic (Logical View).....	6
2.2. Khung nhìn Tiến trình (Process View)	6
2.3. Khung nhìn Cài đặt (Implementation View).....	7
2.4. Khung nhìn Triển khai (Deployment View)	7
2.5. Khung nhìn Ca sử dụng (Use Case View)	7
2.6. Khung nhìn dữ liệu (Data View).....	7
3. Định danh kiến trúc và ràng buộc (Non-Functional Requirements)	8
3.1 Nền tảng kỹ thuật	8
3.2 Giao dịch	8
3.3 Bảo mật	8
3.4 Bền vững	9
3.5 Độ tin cậy/Khả dụng	9
3.6 Hiệu suất.....	9
4. Khung nhìn Ca sử dụng (Use Case View)	9
4.1. Người dùng.....	11
4.2. Hiện thực hóa Ca sử dụng (Use-Case Realization).....	11
5. Khung nhìn Logic (Logical View)	12
5.1 Tổng quan.....	12
5.2. Biểu đồ tuần tự cho quá trình đặt phòng	13
6. Khung nhìn Tiến trình (Process View)	14
7. Khung nhìn Triển khai (Deployment View)	15
8. Hiệu năng chịu tải	15
9. Chất lượng.....	16

Danh mục hình ảnh

Hình 1: Mô hình kiến trúc "4+1"	4
Hình 2: Khung nhìn ca sử dụng	10
Hình 3: Sơ đồ Người dùng	11
Hình 4: Hiện thực hóa ca sử dụng.....	12
Hình 5: Khung nhìn Logic	12
Hình 6: Biểu đồ tuần tự ca sử dụng Đặt phòng	13
Hình 7: Khung nhìn tiến trình	14

Lịch sử sửa đổi

Họ tên	Thời gian	Lý do sửa đổi	Phiên bản
Nguyễn Đức Thiện	08/04/2024	Khởi tạo tài liệu	1.0
Nguyễn Trọng Linh	11/04/2024	Thêm và chỉnh sửa tài liệu	1.1
Nguyễn Trọng Linh	12/04/2024	Thêm và chỉnh sửa tài liệu	1.2
Tô Lâm Sơn	13/04/2024	Chỉnh sửa tài liệu	1.3

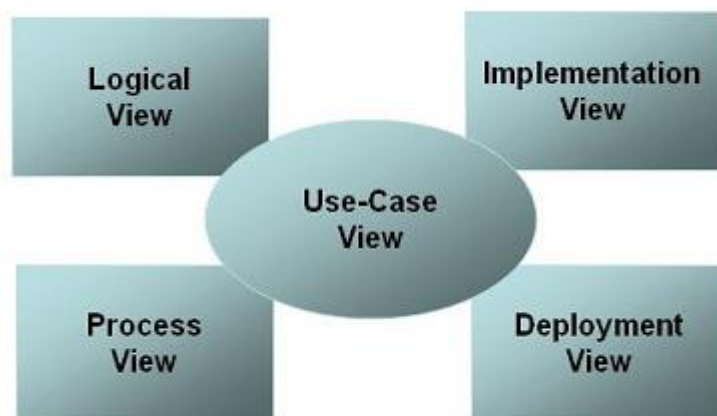
1. Giới thiệu

Tài liệu này cung cấp cái nhìn tổng quan và giải thích toàn bộ quá trình trong hệ thống quản lý khách sạn EzCloud. Tài liệu giải thích cách một người dùng có thể tương tác để thực hiện nhu cầu và bao gồm quy trình kiến trúc cơ bản. Tài liệu cung cấp một mô tả tổng quan về các mục tiêu của kiến trúc, các trường hợp sử dụng được hỗ trợ bởi hệ thống và các phong cách kiến trúc và thành phần đã được chọn để đạt được tốt nhất các trường hợp sử dụng. Khung kiến trúc này sau đó cho phép phát triển các tiêu chí thiết kế và tài liệu xác định các tiêu chuẩn kỹ thuật và lĩnh vực chi tiết.

1.1. Mục đích

Tài liệu Kiến trúc Phần mềm cung cấp tổng quan kiến trúc toàn diện về Hệ thống quản lý khách sạn được cung cấp bởi nhóm sinh viên: Nguyễn Đức Thiện, Tô Lâm Sơn, Mai Tú Phương, Nguyễn Trọng Linh, Phan Xuân Bảo. Tài liệu trình bày một số quan điểm kiến trúc khác nhau để miêu tả các khía cạnh khác nhau của hệ thống, sau đó, được thiết kế để ghi lại và truyền đạt những quyết định kiến trúc quan trọng đã được đưa ra trên hệ thống.

Để miêu tả phần mềm một cách chính xác nhất có thể, cấu trúc của tài liệu này dựa trên mô hình kiến trúc "4+1".



Hình 1: Mô hình kiến trúc "4+1"

Mô hình kiến trúc "4+1" cho phép các bên liên quan khác nhau tìm thấy những gì họ cần trong kiến trúc phần mềm.

1.2. Phạm vi

Phạm vi của SAD này là miêu tả kiến trúc của “Hệ thống quản lý khách sạn” do nhóm sinh viên Nguyễn Đức Thiện, Tô Lâm Sơn, Mai Tú Phương, Nguyễn Trọng Linh, Phan Xuân Bảo xây dựng.

Tài liệu này mô tả các khía cạnh thiết kế của Hệ thống quản lý khách sạn được coi là quan trọng về mặt kiến trúc, tức là, những yếu tố và hành vi được coi là cốt lõi nhất để hướng dẫn xây dựng Hệ thống quản lý khách sạn và để hiểu dự án này một cách toàn thể. Các bên liên quan cần hiểu biết về kỹ thuật của Hệ thống quản lý khách sạn được khuyến khích bắt đầu bằng việc đọc tài liệu này, sau đó xem xét mô hình UML của Hệ thống và sau đó là xem xét mã nguồn.

1.3. Thuật ngữ, Từ đồng nghĩa và Từ viết tắt

- CSSIR: Hệ thống Phục vụ Khách hàng trong nhà hàng (Customer Service System in Restaurant).
- Nginx: máy chủ cân bằng tải và đồng thời cũng là máy chủ web.
- MySQL: hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).
- HTTP: Giao thức truyền tải siêu văn bản (Hypertext Transfer Protocol).
- WWW: mạng lưới toàn cầu (World Wide Web).
- Apache Tomcat: máy chủ App.
- SAD: Tài liệu đặc tả kiến trúc (Software Architecture Document).
- RUP: Quy trình Thống nhất Hợp lý (Rational Unified Process).
- UML: Ngôn ngữ Mô hình hóa Thống nhất (Unified Modeling Language).
- ...

1.4. Tài liệu tham khảo

[PS-Doc]:	HTPVKHTNH, Tài liệu Problem Statement
[UCM-Doc]:	HTPVKHTNH, Tài liệu Use Case Model
[DM-Doc]:	HTPVKHTNH, Tài liệu Domain Model
[SS-Doc]:	HTPVKHTNH, Tài liệu Supplementary Specification
[Glossary-Doc]:	HTPVKHTNH, Tài liệu Glossary
[MedBiquitous]:	Sample SAD, http://medbiq.org/std_specs/techguidelines/softwarearchitecture.pdf
[KRU41]:	The “4+1” view model of software architecture, Philippe Kruchten, November 1995, http://www3.software.ibm.com/ibmdl/pub/software/rational/web/whitepapers/2003/Pbk4p1.pdf

[RUPRSA]: Developing a J2EE Architecture with Rational Software Architect using the Rational Unified Process®, IBM DeveloperWorks, Jean-Louis Maréchaux, Mars 2005, http://www-128.ibm.com/developerworks/rational/library/05/0816_Louis/

1.5. Tổng quan

Phần 2: mô tả việc sử dụng mỗi Khung nhìn.

Phần 3: mô tả các ràng buộc kiến trúc của hệ thống.

Phần 4: mô tả các Yêu cầu Chức năng ảnh hưởng đáng kể đến kiến trúc.

Phần 5: mô tả cách thực hiện Ca sử dụng hiện thực hóa quan trọng nhất.

Phần 6: mô tả các khía cạnh mang tính đồng thời của thiết kế.

Phần 7: mô tả cách hệ thống sẽ được triển khai như thế nào.

Phần 8: mô tả yếu tố lưu trữ quan trọng.

Phần 9: mô tả vấn đề về hiệu suất và ràng buộc.

Phần 10: mô tả khía cạnh liên quan đến các thuộc tính chất lượng dịch vụ (QoS).

2. Biểu diễn kiến trúc

Tài liệu này mô tả kiến trúc của “Hệ thống quản lý khách sạn EzCloud” bằng cách sử dụng các Khung nhìn được định nghĩa trong mô hình "4+1" và sử dụng các quy ước đặt tên trong phương pháp RUP. Các kiểu nhìn được sử dụng để tài liệu hóa hệ thống trên bao gồm:

2.1. Khung nhìn Logic (Logical View)

- Đối tượng sử dụng: Người thiết kế.
- Phạm vi:
 - Yêu cầu chức năng: mô tả mô hình đối tượng của thiết kế. Đồng thời mô tả các trường hợp sử dụng quan trọng nhất và các yêu cầu nghiệp vụ của hệ thống.
 - Các tài liệu liên quan: mô hình thiết kế.

2.2. Khung nhìn Tiến trình (Process View)

- Đối tượng sử dụng: Người tích hợp.
- Phạm vi:
 - Yêu cầu phi chức năng: mô tả các khía cạnh đồng thời và đồng bộ hóa của thiết kế.
 - Các tài liệu liên quan: không có mô hình cụ thể.

2.3. Khung nhìn Cài đặt (Implementation View)

- Đối tượng sử dụng: Lập trình viên.
- Phạm vi:
 - Thành phần phần mềm: mô tả các lớp và các hệ thống con của hệ thống.
 - Tài liệu liên quan: mô hình triển khai, mô hình thành phần.

2.4. Khung nhìn Triển khai (Deployment View)

- Đối tượng sử dụng: Quản lý triển khai hệ thống.
- Phạm vi:
 - Tô pô học: mô tả cách phân bố phần mềm trên phần cứng và hiển thị các khía cạnh phân phối của hệ thống. Mô tả các cấu trúc triển khai tiềm năng bằng cách bao gồm các kịch bản triển khai đã biết và dự kiến trong kiến trúc để cho phép người triển khai đưa ra một số giả định về hiệu suất, tương tác hệ thống và các vấn đề khác.
 - Các tài liệu liên quan: Mô hình triển khai.

2.5. Khung nhìn Ca sử dụng (Use Case View)

- Đối tượng sử dụng: Tất cả các bên liên quan đến hệ thống, bao gồm cả Người dùng cuối.
- Phạm vi:
 - Mô tả tập hợp các kịch bản, các Ca sử dụng biểu diễn một số chức năng quan trọng của hệ thống. Mô tả các Người và Ca sử dụng hệ thống, khung nhìn này trình bày nhu cầu của người dùng và mở rộng thêm ở mức thiết kế để mô tả các luồng và ràng buộc rõ ràng hơn.
 - Các tài liệu liên quan: Mô hình Ca sử dụng, tài liệu Ca sử dụng.

2.6. Khung nhìn dữ liệu (Data View)

- Đối tượng sử dụng: Chuyên gia dữ liệu, Quản trị cơ sở dữ liệu
- Phạm vi:
 - Bền vững: mô tả các yếu tố cốt lõi về yếu tố bền vững trong mô hình dữ liệu
 - Các tài liệu liên quan: Mô hình dữ liệu.

3. Định danh kiến trúc và ràng buộc (Non-Functional Requirements)

Phần này mô tả các yêu cầu và mục tiêu phần mềm có tác động đáng kể đến kiến trúc.

3.1 Nền tảng kỹ thuật

Máy chủ

Máy chủ hệ thống sẽ được lưu trữ trên máy chủ Web nội bộ Nginx và máy chủ App nội bộ Apache Tomcat tại hệ thống và kết nối với máy chủ MySQL. Máy chủ Nginx sẽ chấp nhận tất cả các yêu cầu từ người dùng và chuyển tiếp chúng đến máy chủ App và máy chủ App chuyển tiếp chúng tới máy chủ Cơ sở dữ liệu. Khi thanh toán, tất cả các giao tiếp với người dùng phải tuân thủ các tiêu chuẩn giao tiếp công khai HTTPS, TCP/IP.

Phía người dùng

Người dùng chỉ có thể truy cập hệ thống thông qua ứng dụng Android trên máy tính bảng do nhà hàng cung cấp.

3.2 Giao dịch

Ứng dụng trực tuyến là có tính giao dịch, tận dụng các tiện ích thanh toán bên ngoài như PayPal, thẻ quốc tế, ví điện tử Momo, ...

3.3 Bảo mật

Do kết nối nội bộ nên giao thức HTTP sẽ được sử dụng để tạo điều kiện cho việc truyền thông giữa máy khách và máy chủ. Mặc dù sẽ sử dụng các cơ chế xác thực mật khẩu cơ bản và phân quyền dựa trên vai trò để bảo vệ hệ thống khỏi truy cập trái phép, chức năng như giao dịch tiền. Hệ thống phải được bảo mật để khách hàng có thể thực hiện thanh toán trực tuyến.

Ứng dụng phải thực hiện các hành vi bảo mật cơ bản:

- Xác thực: Đăng nhập bằng tên người dùng và mật khẩu
- Phân quyền: Theo các thông số kỹ thuật phần mềm, quản trị viên sẽ có đặc quyền nâng cao để thực hiện các nhiệm vụ mà người dùng thông thường không được ủy quyền.

Đối với truy cập internet, các yêu cầu sau là bắt buộc:

- Bảo mật thông tin: Dữ liệu nhạy cảm phải được mã hóa (thanh toán bằng thẻ tín dụng)
- Tính toàn vẹn dữ liệu: Dữ liệu được gửi qua mạng không thể bị thay đổi bởi một lớp nào đó.

3.4 Bền vững

Sự bền vững dữ liệu sẽ được giải quyết bằng cách sử dụng cơ sở dữ liệu quan hệ.

3.5 Độ tin cậy/Khả dụng

- Khả năng mở rộng và độ tin cậy của hệ thống là yêu cầu chính vì liên quan đến tính chất của hệ thống này.
- Không gian máy chủ phải đáp ứng được khối lượng dữ liệu và lưu lượng truy cập ngày càng tăng. Độ tin cậy/Khả dụng sẽ được giải quyết thông qua nền tảng máy chủ.
- Mục tiêu khả dụng là 15/7: 15 giờ một ngày, 7 ngày một tuần.
- Thời gian còn lại (9 giờ) được dành cho các hoạt động bảo trì.
- Trong trường hợp máy chủ bị lỗi có thể dẫn đến hệ thống tạm thời không khả dụng. Kỳ vọng về hiệu suất là nếu bất kỳ thao tác nào mất hơn 15 giây để thực hiện, nó sẽ được thực hiện như một thao tác nền, giải phóng người dùng để thực hiện các tác vụ khác. Người dùng phải được thông báo về tất cả các thao tác với phản hồi hình ảnh.

3.6 Hiệu suất

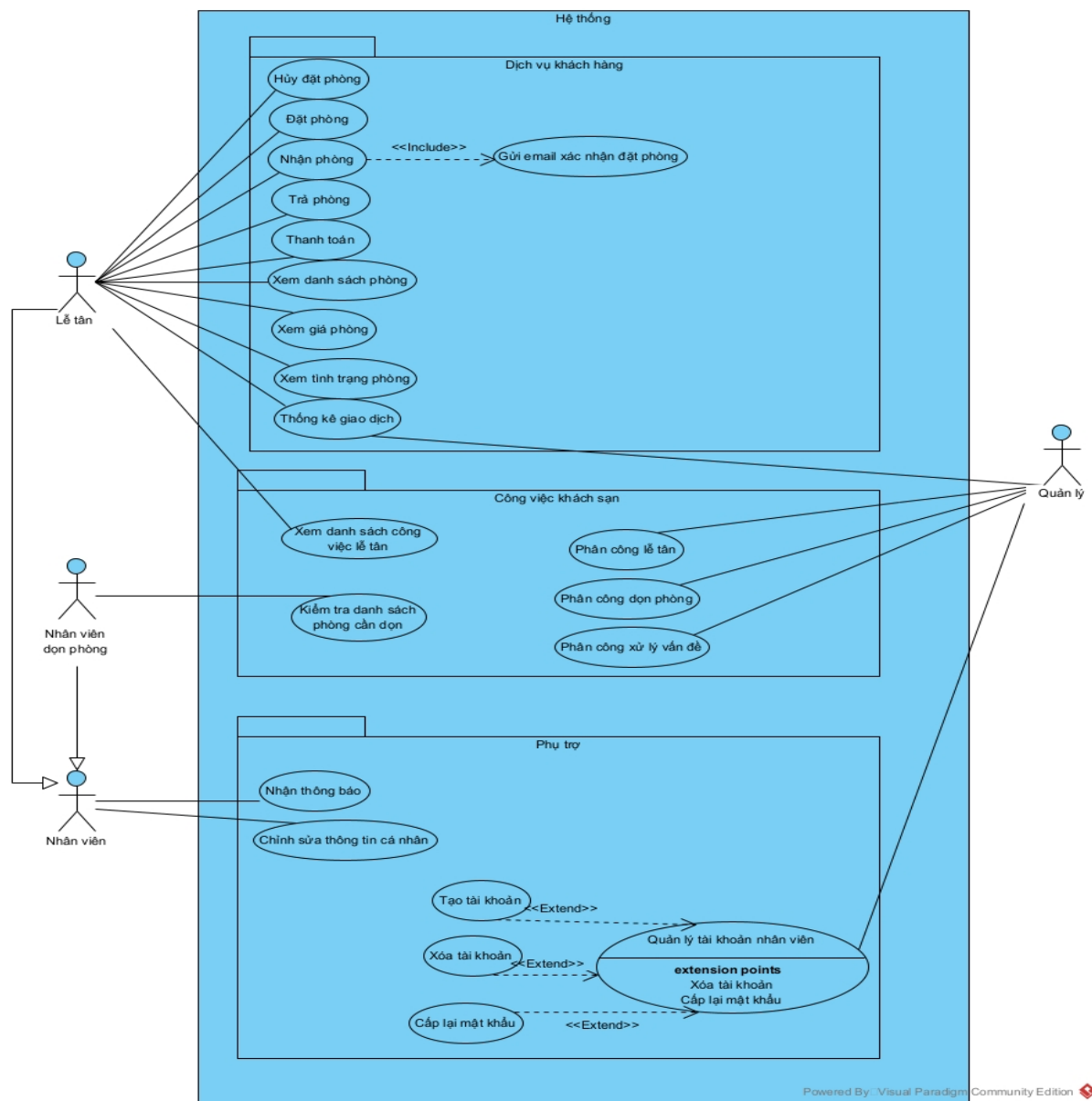
Quá trình thanh toán (xác nhận và xác nhận thẻ tín dụng) phải dưới 10 giây trên máy chủ Apache Web.

4. Khung nhìn Ca sử dụng (Use Case View)

Dưới đây là danh sách các Ca sử dụng từ tài liệu đặc tả Ca sử dụng (Use Case Model) mà đại diện cho các chức năng quan trọng và có độ ưu tiên cao nhất của hệ thống. Các Ca sử dụng ảnh hưởng đáng kể tới kiến trúc trong [UC-Doc] là:

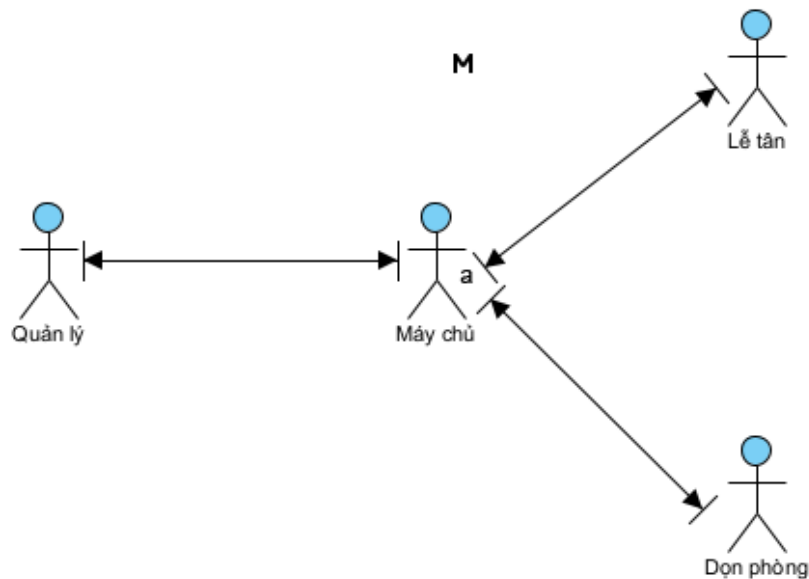
1. Đặt phòng
2. Hủy đặt phòng
3. Gửi email xác nhận đặt phòng
4. Nhận phòng
5. Trả phòng
6. Thanh toán
7. Xuất hóa đơn
8. Xem danh sách khách hàng
9. Thống kê các giao dịch
10. Phân chia công việc cho lễ tân
11. Phân chia công việc cho dọn phòng
12. Xử lý vấn đề phát sinh

13. Quản lý tài khoản nhân viên



Hình 2: Khung nhìn ca sử dụng

4.1. Người dùng



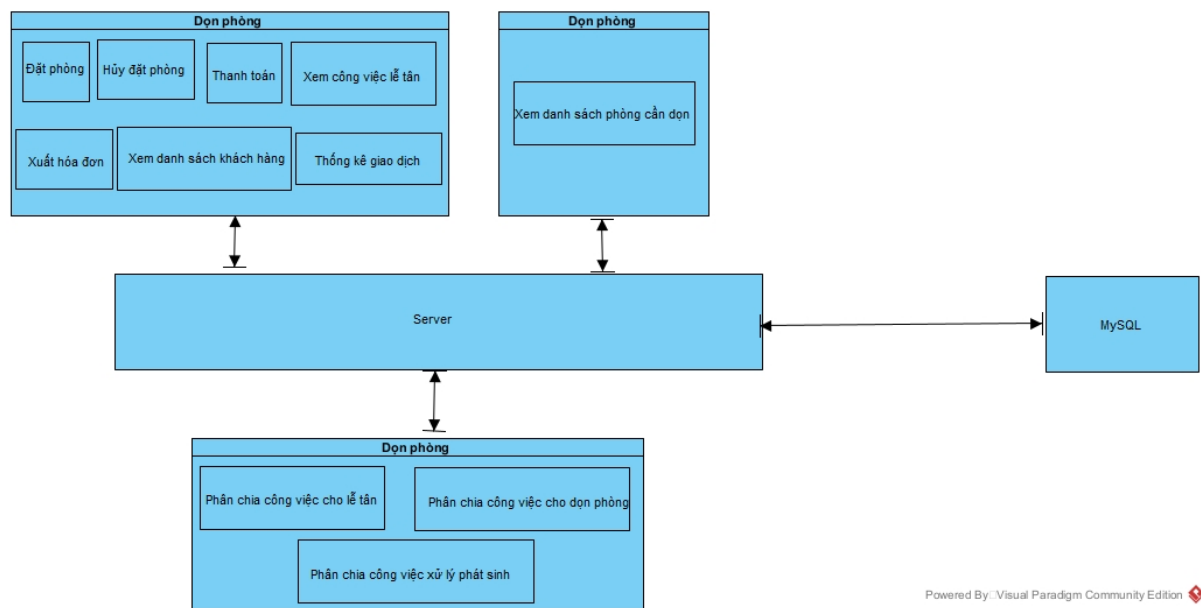
Hình 3: Sơ đồ Người dùng

Người dùng chính (Primary Actor) có thể đóng một trong các vai trò sau

- Quản lý
- Lễ tân
- Dọn phòng
- Máy chủ: là Người dùng loại 3 (3rd type) và cũng chính là hệ thống CSSIR. Nó xử lý tất cả các tiến trình vật lý và logic của phần mềm trong hệ thống.

4.2. Hiện thực hóa Ca sử dụng (Use-Case Realization)

Sơ đồ chức năng của Ca sử dụng bên dưới mô tả cách các Yếu tố Thiết kế (Design Element) cung cấp các chức năng được xác định trong các Ca sử dụng quan trọng. Trong sơ đồ này, Người dùng chung được giả định là cùng một người với Khách chưa có tài khoản và Khách đã có tài khoản. Các Ca sử dụng được hiển thị như các chức năng cho hệ thống. Chức năng có thể bao gồm nhiều hơn một Ca sử dụng. Được giả định rằng đăng nhập là chức năng mặc định và nó phải được thực hiện trước khi bất kỳ chức năng nào khác sẽ được kích hoạt.

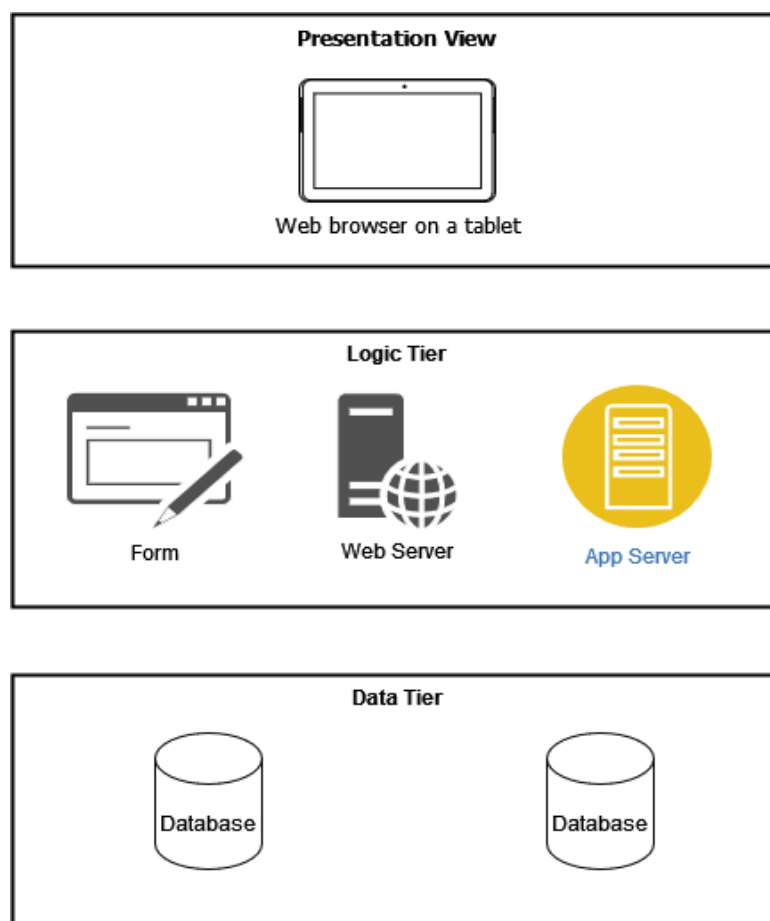


Hình 4: Hiện thực hóa ca sử dụng

5. Khung nhìn Logic (Logical View)

5.1 Tổng quan

“Hệ thống quản lý khách sạn” được chia thành các tầng dựa trên kiến trúc 3-tier.



Hình 5: Khung nhìn Logic

Mô hình phân tầng của hệ thống dựa trên một chiến lược phân bổ trách nhiệm liên quan mỗi tầng với một trách nhiệm cụ thể.

Chiến lược này được chọn vì chiến lược này cách ly các trách nhiệm hệ thống khác nhau với nhau, giúp cải thiện phát triển và bảo trì hệ thống.

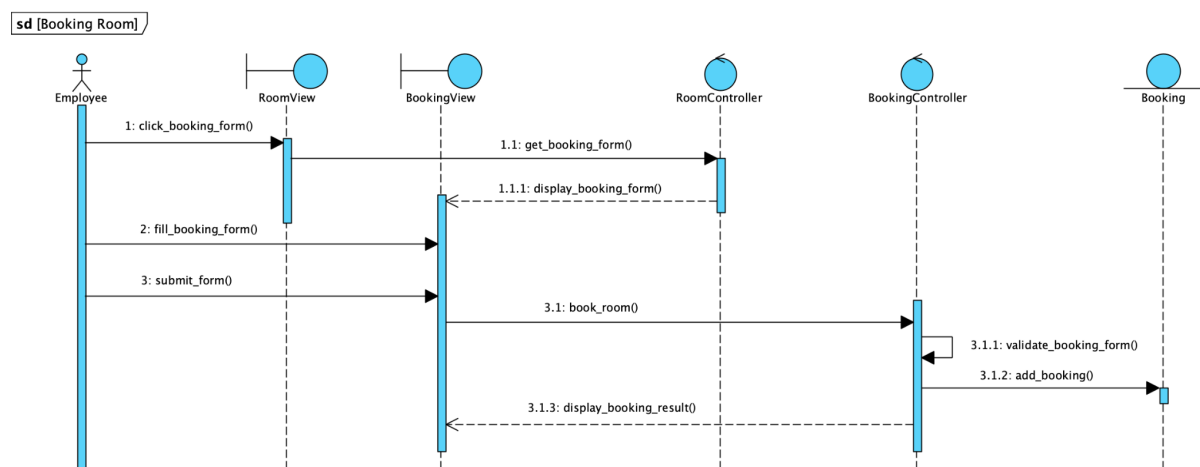
Các trách nhiệm cụ thể của mỗi tầng như sau:

- Tầng trình bày (Presentation Layer) xử lý logic trình bày và giao diện hiển thị trên hệ thống. Tầng trình bày chứa tất cả các thành phần cần thiết để cho phép tương tác với người dùng cuối. Nó bao gồm giao diện người dùng. Các bản xem trước của giao diện được đặt trong Tài liệu Thiết kế Phần mềm.
- Tầng logic (Logic Layer) xử lý các yêu cầu do người dùng cuối gửi cho tầng trình bày. Khi cần tài nguyên, nó sẽ lấy tài nguyên từ tầng dữ liệu để có thể trả lại cho tầng trình bày những phản hồi mà tầng trình bày cần.
- Tầng dữ liệu (Data Layer) chứa các máy chủ cơ sở dữ liệu cũng như tài nguyên thanh toán để tầng logic có thể tác động vào và truy vấn những dữ liệu hợp lý, phục vụ cho quá trình tạo ra phản hồi gửi lại cho người dùng cuối.

5.2. Biểu đồ tuần tự cho quá trình đặt phòng

Biểu đồ tuần tự cho quá trình đặt phòng:

1. Nhân viên mở form đặt phòng.
2. Hệ thống hiển thị form đặt phòng.
3. Nhân viên điền các thông tin được yêu cầu trong form đặt phòng.
4. Nhân viên gửi form và hệ thống sẽ xử lý yêu cầu đặt phòng.
5. Hệ thống thông báo kết quả đặt phòng.



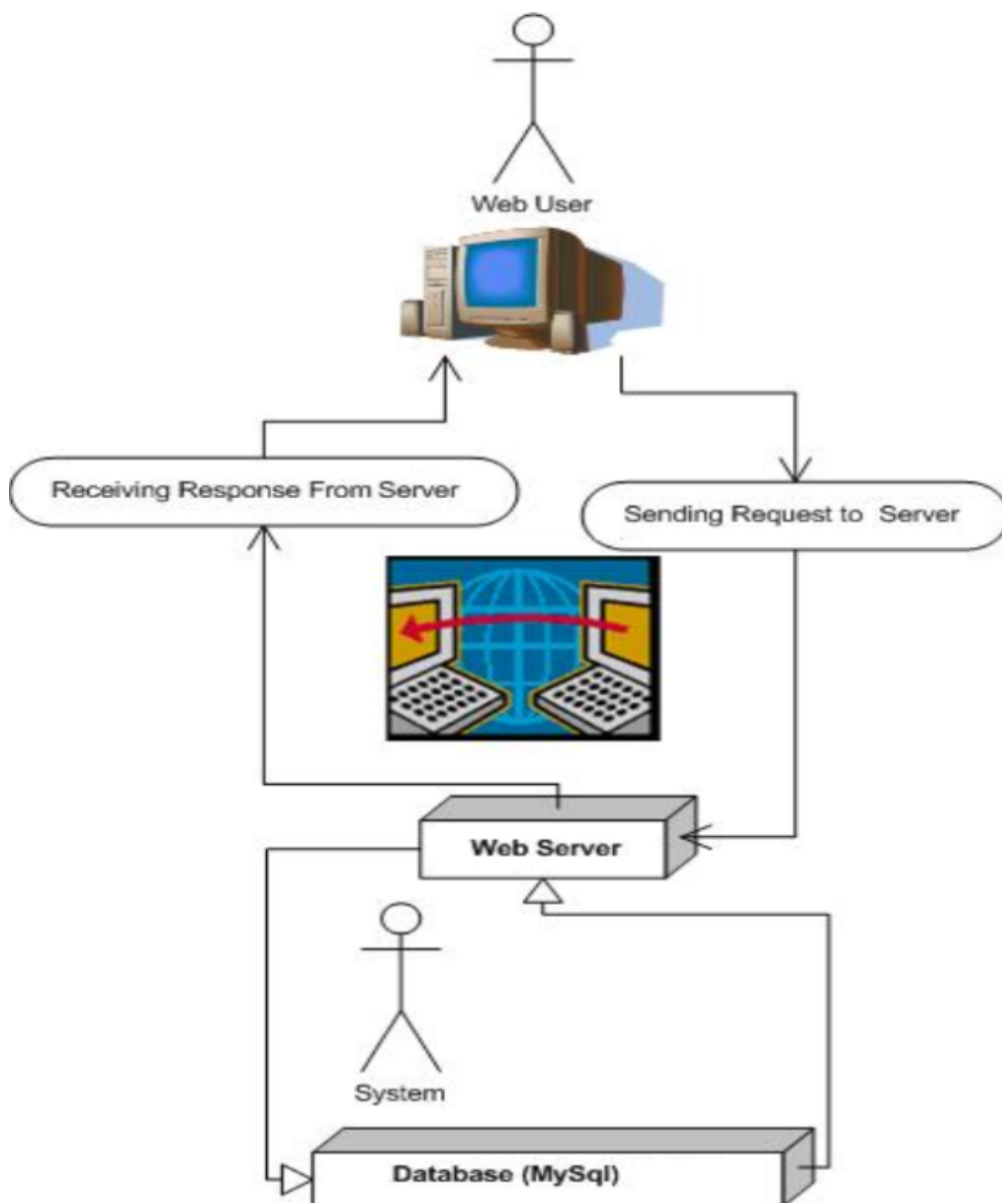
Hình 6: Biểu đồ tuần tự ca sử dụng Đặt phòng

6. Khung nhìn Tiến trình (Process View)

Chỉ có một quy trình cần tính đến. Máy sẽ tự động xử lý các luồng là phiên bản của quy trình này.

Sơ đồ dưới đây mô tả các vòng tròn quy trình. Có hai vòng tròn quy trình:

Người dùng – vòng kết nối máy chủ web và vòng kết nối cơ sở dữ liệu máy chủ web. Yêu cầu từ người dùng trước tiên sẽ chuyển đến máy chủ web. Trước tiên, máy chủ web đánh giá yêu cầu theo quy tắc/yêu cầu nghiệp vụ và xác định xem có cần thiết lập kết nối với cơ sở dữ liệu hay không. Nếu kết nối là cần thiết, điều đó được hoàn thành trước và chỉ sau đó người dùng mới được trả về với phản hồi từ máy chủ web.



Hình 7: Khung nhìn tiến trình

7. Khung nhìn Triển khai (Deployment View)

Tổng quan toàn cục

Cấu trúc phần cứng:

- Máy chủ:
 - Loại máy chủ: Cấu hình máy chủ sẽ phụ thuộc vào số lượng người dùng, lượng truy cập và dữ liệu của hệ thống.
 - Hệ điều hành: Linux (Ubuntu Server, CentOS) hoặc Windows Server.
 - Vị trí: Đặt tại trung tâm dữ liệu (data center) để đảm bảo tính an toàn, tin cậy và khả năng mở rộng.
- Thiết bị người dùng:
 - Điện thoại thông minh (smartphone)
 - Máy tính bảng (tablet)
 - Máy tính để bàn (desktop)

Phần mềm:

- Nền tảng: Điện toán đám mây (cloud computing) để đảm bảo tính linh hoạt và khả năng mở rộng.
- Ứng dụng: Ứng dụng web.
- Ngôn ngữ lập trình: AngularJS, jQuery,...
- Công nghệ:
 - Cloudflare: Để tăng tốc độ tải trang web và bảo mật hệ thống.

Bảo mật:

- Để đảm bảo tính toàn vẹn và bảo mật của dữ liệu, hệ thống sẽ sử dụng các phương thức bảo mật như mã hóa SSL (Secure Sockets Layer) để mã hóa dữ liệu giao tiếp.
- Sử dụng JWT (JSON Web Token) để xác thực, xác minh phiên người dùng.

8. Hiệu năng chịu tải

Kích thước

- Hỗ trợ tối thiểu 10000 lượt truy cập.
- Hỗ trợ tối thiểu 20000 tương tác đồng thời.
- Ứng dụng cần đáp ứng đồng thời được ít nhất 50 nhân viên sử dụng.

- Dự kiến có khoảng ~3000 người dùng đã đăng ký sử dụng ứng dụng.
- Truy cập tới cơ sở dữ liệu và phản hồi chậm không quá 5 giây.

Xử lý thao tác của các nhân viên trên hệ thống nhanh

- Truy cập tới server cơ sở dữ liệu và phản hồi với thời gian nhỏ hơn 5 giây.
- Hoàn thành 95% yêu cầu của khách hàng trong chưa đầy 30 giây.

9. Chất lượng

Đối với hệ thống quản lý khách sạn, các mục tiêu chất lượng sau đã được xác định:

Khả năng mở rộng:

- Mô tả: Hệ thống cần phản ứng linh hoạt khi có sự gia tăng đột ngột trong số lượng yêu cầu từ người dùng.
- Giải pháp: Sử dụng các máy chủ hệ thống hỗ trợ một số kỹ thuật quản lý khối lượng công việc để đảm bảo hiệu suất vận hành.

Đáng tin cậy và khả dụng:

- Mô tả: Đảm bảo sự liên tục hoạt động mà không gây ra thời gian ngừng hoạt động.
- Hướng giải quyết: Sử dụng các cơ chế tự động phát hiện và chuyển đổi dự phòng để tránh sự cố và giảm thiểu thời gian ngừng hoạt động.

Di động:

- Mô tả: Hệ thống cần có khả năng triển khai và sử dụng trên môi trường khác nhau một cách dễ dàng.
- Giải pháp: Đảm bảo hệ thống tuân thủ đầy đủ các yêu cầu cơ bản và có thể triển khai trên các máy chủ web hoặc máy chủ địa phương.

Bảo mật:

- Mô tả: Yêu cầu các cơ chế xác thực và phân quyền mạnh mẽ để đảm bảo an toàn thông tin của khách hàng và hệ thống.
- Giải pháp: Sử dụng lại các cơ chế bảo mật cục bộ đã được kiểm chứng để tăng cường tính bảo mật của hệ thống.